

RECEIVED  
CENTRAL FAX CENTER

AUG 30 2007

COMMENTS

The enclosed supplemental amendment is responsive to conversations held with Examiner Kendall on 6/18/07 and 8/29/07 concerning the allowance of the instant case and related application 10/750,050 (the " '050 application"). During these conversations Examiner Kendall expressed a willingness to allow the present application and the '050 application subject to the filing of a terminal disclaimer that merged the term and ownership of the two applications.

In both conversations the applicant expressed a willingness to file a terminal disclaimer between these two applications. The applicant also expressed a willingness to file terminal disclaimers against other related cases (e.g., application 10/749,617) *if doing so would advance any such applications towards allowance*. During the 6/18/07 conversation the Applicant expressed a desire to receive an Office Action from the Examiner in the instant case so that, should pertinent art arise in other related applications, they could be identified to the Examiner.

Consistent with statements made by the Applicant during the 6/18 and 8/29 phone calls, the Applicant has included a terminal disclaimer that merges the instant application with the '050 application, but, has not submitted any other terminal disclaimers. With respect to the terminal disclaimer presently being filed, the applicant notes that (contrary to views expressed to the Examiner by the Applicant during the 8/29/07 phone call), the claims of the present application and the claims of the '050 application appear to be patentably distinct from one another at least because the claims of the present application recite the use of two different

dictionaries for the same instrumented method (which the claims of the '050 application do not recite), while, the claims of the '050 application recite the passing of classfile name, method name and argument types during invocation by an instrumented method (which the claims of the present application do not recite).

#### Updated Information Disclosure Statement

The Applicant also submits herewith an updated Information Disclosure Statement (IDS). Amongst the references cited in the enclosed 1449 form, the Applicant believes the Boykin reference is the most pertinent and warrants some discussion so that the Examiner can scrutinize his decision to allow the present application. The Boykin reference, like the Applicant's specification, is directed to a bytecode modification system. However, the Boykin reference is different from the subject matter claimed by the present application in at least two respects. First, the Boykin reference appears to be limited to disclosing a method that only instruments a classfile with additional bytecode instructions during the loading of the classfile at runtime, while, the claims of the present application clearly recite that bytecode modification of a classfile occurs prior to the loading of the classfile (i.e. before runtime). Second, the Boykin reference does not appear to disclose that two different dictionaries are used during a dispatching process executed from the same instrumented method. The Applicant's claims specifically recite this feature.

With respect to the former point made above (that Boykin is directed to bytecode modification during classloading), the Examiner is invited to compare, the

following excerpts of Boykin (where "hooks" are understood to be bytecode modifications)

At class load time, an injector determines whether a loaded class has any instrumentation locations as predetermined by information in the registry. If so, the injector inserts hooks in the loaded class.

Boykin, abstract (emphasis added).

Using the probe location information, injector component 214 inserts hooks into original class files 202 at class load time to create manageable classes 216 comprising hooks . . .

Boykin, para. [0037] (emphasis added).

During the class load process, the class loader provides an indication, e.g., class load event notification 408, to injector 410, which then injects hooks into the classes.

Boykin, para. [0045] (emphasis added).

with the following comments that appear in the Applicant's specification

. . . the bytecode modifier module 352 may modify the bytecode 350 prior to runtime . . . Applicant's Specification, para. [0053] (emphasis added).

According to the depiction of Figure 17, the method associated with each of information structures 1703<sub>1</sub> through 1703<sub>N</sub> "registers" with the dispatch unit 1702 by sending the dispatch unit 1702 its class name, the names of each of its methods and the types of arguments of its methods. By way of example, as illustrated in Figure 17, modified classfile 1701<sub>1</sub> sends the dispatch unit 1702 its class name and the names and argument types of each of its method 1705 . . . In an embodiment registration occurs as a consequence of a modified classfile being loaded. Applicant's Specification, paras. [0144], [0145], [0146] (emphasis added).

Comparison of these excerpts clearly reveals that the Boykin reference discloses actual modification of byte code at runtime during classloading while the Applicant's specification discloses actual modification of byte code before runtime and therefore prior to classloading. The Applicant notes that the Avakian reference (already made of record in the present application and discussed by the Applicant in

a previous Office Action), like Boykin, appears to instrument classfiles during classloading rather than prior to classloading.

With respect to the second point made above (that the Boykin fails to disclose the use of two different dictionaries during dispatching from the same instrumented method), the Applicant notes that during runtime of the system taught by Boykin an instrumented method is described as only invoking a "registry". The Boykin does not appear to describe the use of more than one registry when "the registry" is invoked by an instrumented method - not does it appear to describe that, after being invoked by an instrumented method, two separate information resources inside the registry are used for the same instrumented method. The following excerpts from Boykin bear this point out.

Using the probe location information, injector component 214 inserts hooks into original class files 202 at class load time to create manageable classes 216 comprising hooks 218, which read probe information 208 from the registry 206 in accordance with an association between the location of a particular hook and one or more probes that are associated with that location. Thereafter, hooks 218 execute probes 220.

Boykin, para. [0037] (emphasis added)

When a hook is executed, it gets the list of probes that are currently enabled for its location from the registry and executes them.

Boykin, para. [0038] (emphasis added)

Similarly, when a hook is invoked, the hook can determine whether a probe is enabled for its location by querying the registry, e.g., by providing an identifier for the location in which the hook was embedded. If the registry has an enabled probe, the hook gets and executes probe 418.

Boykin, para. [0048] (emphasis added)

Assuming there is at least one enabled probe, the hook queries the registry for the probes (step 804). The hook then executes the probes (step 806), and the probe runtime process is complete.

Boykin, para. [0052] (emphasis added).

The Applicant's specification, by contrast, specifically indicates in paragraph [0097] that different dictionaries could be used for entry and exit methods which, as described throughout the Applicant's specification, can be inserted into the same instrumented method.

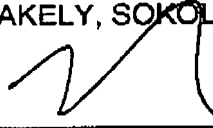
Thus the Applicant believes the present claims are allowable over the newly disclosed Boykin reference and respectfully requests allowance of the present application as a whole.

If there are any additional charges, please charge Deposit Account No. 02-2666. If a telephone interview would in any way expedite the prosecution of this application, the Examiner is invited to contact Robert B. O'Rourke at (408) 720-8300.

Respectfully submitted,

BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Dated: 8/30/07

  
Robert B. O'Rourke  
Reg. No. 46,972

12400 Wilshire Blvd.  
Seventh Floor  
Los Angeles, CA 90025-1030  
(408) 720-8300